

LISTING OF AND AMENDMENTS TO CLAIMS:

1. (currently amended) A method of selectively displaying a markup language form element on a Web page when an object is available to a Web server, the method comprising the steps of:

(a) associating the markup language form element with the object;

(b) reading a configuration file to determine determining whether the object is available to the Web server;

(c) generating a markup language document that includes the markup language form element if the object is available to the Web server; and

(d) displaying the markup language form element on the Web page if the markup language document contains the markup language form element.

2. (original) The method of claim 1, wherein the markup language is HTML.

3. (original) The method of claim 1, wherein the markup language form element is a button.

4. (currently amended) The method of claim 1, wherein step (b) includes maintaining a record of whether the object is available to the Web server within said [[a]] configuration file.

5. (original) The method of claim 4, wherein the configuration file is written in XML.

6. (original) The method of claim 1, wherein the object is an application program.

7. (original) The method of claim 1, wherein step (c) further comprises the step of generating a markup language document that does not include the markup language form element if the object is not available to the Web server.

8. (original) The method of claim 1, wherein step (c) further comprises the step of generating a markup language document that includes a disabled markup language form element if the object is not available to the Web server.

9. (original) A computer program comprising means adapted to perform all the steps of claim 1 when said program is run on a networked computer.

10. (original) A computer program as claimed in claim 9 embodied on a computer readable medium.

11. (currently amended) A program product stored on a computer readable medium for generating a markup language document for display on a Web page when an object is available to a Web server, said program product comprising:

(a) means for associating the markup language form element with the object;

(b) means for reading a configuration file to determine ~~determining~~ whether the object is available to the Web server;

(c) means for generating a markup language document that includes the markup language form element if the object is available to the Web server; and

(d) means for displaying the markup language form element on the Web page if the markup language document contains the markup language form element.

12. (original) The product of claim 11, wherein the markup language is HTML.

13. (original) The product of claim 11, wherein the markup language form element is a button.

14. (currently amended) The product of claim 11, further comprising means for maintaining a record of whether the object is available to the Web server within said [[a]] configuration file.

15. (original) The product of claim 14, wherein the configuration file is written in XML.

16. (original) The product of claim 11, wherein the object is an application program.

17. (original) The product of claim 11, further comprising means for the step of generating a markup language document that does not include the markup language form element if the object is not available to the Web server.

18. (original) The product of claim 11, further comprising means for generating a markup language document that

includes a disabled markup language form element if the object is not available to the Web server.

19. - 27. (canceled).

28. (currently amended) A computer system for selectively displaying a markup language form element on a Web page when an associated object is available to a Web server, said computer system comprising:

(a) a server for generating a markup language document, said server comprising:

(i) storage means for storing a configuration file having a value representing whether the object is available to the Web server;

(ii) a first processing means coupled to said storage means for:

(A) determining whether the object is available to the Web server;

(B) generating a markup language document that includes the markup language form element if the object is available to the Web server;

(b) a communications network coupled to said server;

and

(c) a second processing means coupled to said server through said communications network, said second processing means for reading and displaying the markup language document.

29. (original) The system of claim 28, wherein the markup language is HTML.

30. (original) The system of claim 28, wherein the markup language form element is a button.

31. (original) The system of claim 28, wherein said first processing means further generates a markup language document that does not include the markup language form element if the object is not available to the Web server.

32. (original) The system of claim 28, wherein said first processing means further generates a markup language document that includes a disabled markup language form element if the object is not available to the Web server.

33. - 37. (Canceled).

38. (new) The method of claim 1, wherein said object is a program associated with a suite of programs which may be installed or uninstalled, and said configuration file indicates whether said program is installed.

39. (new) The method of claim 1, further comprising updating said configuration file periodically to determine availability of at least one of the object and other objects.

40. (new) The method of claim 1, wherein step (a) is performed prior to step (d).

41. (new) The method of claim 1, further comprising creating a hash file of data stored in said configuration file for use in determining whether to display the markup language form element.

42. (new) The method of claim 1, wherein if the configuration file does not contain a reference to an object, the element associated with the object will be displayed.

43. (new) The product of claim 11, wherein said object is a program associated with a suite of programs which may be installed or uninstalled, and said configuration file indicates whether said program is installed.

44. (new) The product of claim 11, further comprising means for updating said configuration file periodically to determine availability of at least one of the object and other objects.

45. (new) The product of claim 11, further comprising means for creating a hash file of data stored in said configuration file for use in determining whether to display the markup language form element.

46. (new) The product of claim 11, wherein if the configuration file does not contain a reference to an object, the element associated with the object will be displayed.

47. The product of claim 11, in combination with a server system, the server system being for implementing operation of said product.

48. (new) The system of claim 28, wherein said object is a program associated with a suite of programs which may be installed or uninstalled, and said configuration file indicates whether said program is installed.

49. (new) The system of claim 28, further comprising means for updating said configuration file periodically to determine availability of at least one of the object and other objects.

50. (new) The system of claim 28, further comprising means for creating a hash file of data stored in said configuration file for use in determining whether to display the markup language form element.

51. (new) The system of claim 11, wherein if the configuration file does not contain a reference to an object, the element associated with the object will be displayed.